Dynamic Language Models for Exploring Early Modern English Texts (1473-1800)

Brian Kitano

2 May 2019

Abstract

The emergence of modern English is rooted in early writings and publications at the advent of the printing press, starting in the 16th and 17th centuries. Previous work has investigated the change in proportion of topics and keywords represented in publications during this time using the Early English Books Online (EEBO) text corpus of documents from 1473 to 1820. We build upon this work by analyzing the semantic change of words and topics represented in the literature via dynamic word embeddings and dynamic topic models. First, we demonstrate that dynamic word vectors appear to reflect shifts in word usage well, alongside speaker-wide historical events and emergent cultural phenomena. Second, we show that dynamic topic models are sensitive to corpus quality. Finally, we provide researchers with the ability to search through the corpus by topic or word semantics. This may be helpful for sociologists and historians to more expressively search and analyze texts of the era.

1 Introduction

The emergence of the modern Western economic thought began in Europe following the height of the Enlightenment. In the short period from 1473 to 1700, events like the discovery of the New World and the beginning of the era of colonization, the advent of the printing press, the invention of the corporation, and the rise of the merchant in the public sphere dramatically transformed society. The public's discourse not only began to include topics like trade and finance, but ideas such as politics and nationhood were transformed by the adoption of economic ideas. Alongside this shift in the discourse itself, the volume of publications grew dramatically as the cost of disseminating information decreased.

Understanding the semantics of words during that era enables a fuller and more accurate depiction of events as they happened. In addition, being able to analyze drifts in the meaning of words from one time period to another provides an understanding of how society changed its conception of these phenomena.

The Early English Books Online (EEBO) collection is a machine-readable corpus of roughly 60,000 texts written from 1473 to 1700; around 2300 of the

documents are "economic" in nature. [22] Natural language processing (NLP) offers a wide variety of statistical tools that can be used to digest and summarize the large body of work that was created by both individuals and institutions writing at the time. Previous work in NLP analysis of the corpus demonstrated the shift in the proportions of topics over time; for example, 'husbandry' reaches a peak in topic proportion at around 1590, declines to a low at around 1620, and remains low for the duration of the period. [10]

Here, we seek to identify temporal shifts in the semantics of individual words and in the topics present in the time period by using dynamic word embeddings and dynamic topic models. Unlike in traditional topic modeling, where we assume exchangeability between the documents present in the corpus, dynamic topic models seek to infer time-based dependencies given an set of documents ordered in time. Similarly, while in traditional word embeddings words and their contexts are treated without consideration for semantic drift, temporal word embeddings seek to represent semantic changes in words by aligning their embeddings with their usage in a given set of time intervals.

2 Historical Context

The three and a half centuries from 1473 to 1820 were an extremely vast and significant time in modern English history. At the start of this era, the New World had not yet been discovered, the Church of England had not yet been formed, and the printing press had been used for the first time on an English text that year.[9] By the end, England had fought and lost both the Revolutionary War and the War of 1812, the East India Company had established a global dominance on trade, and the written word was an invisible part of English life. Literacy among men rose from an estimated 10% in 1500 to 65% by 1820.[17] In the same period, printed publications in England grew from approximately 11,000 books in the 16th century to 230,000 in the 18th century. [19]

The impact of historical world events alongside the democratization of written text due to the printing press led to dramatic changes in the usage and meaning of English, as well as the public discourse of English readers and writers. For example, the topic 'war' should shift from the English invasion of France and Scotland in 1514, to war with Spain in the late 1500s, to colonial unrest in the Americas in the middle of the 18th century. Similarly, we hypothesize that the semantics of a word like 'trade' would shift to have more colonial connotations towards the end of the era, as the British Empire grew dramatically from 1600 onwards.[23]

3 Previous Work

In [24], the authors aim to identify how 'concepts,' or groups of words, evolve over time, based on word embeddings, called Concepts Through Time. This seems like the most promising work, but their model is not provided in their

publication. They apply their model to trace semantic shifts in Dutch public discourse from 1890 to 1990 surrounding consumerism, globalization, and economic models.

In [16], the authors provide specific tools for working with EEBO, including "the management of orthographic variance" and "the ability to create specially-tailored subsets of the EEBO-TCP corpus based on criteria such as date, title keyword, or author." While this is helpful in providing researchers with important EEBO-TCP metadata, it doesn't provide semantic models of the corpus or topic tagging of documents.

In [10], the authors aim to study the correlation between the publication of economic documents and the emergence of economic phenomena, including chartered companies and rates of merchant representation, from 1550 to 1720. To perform their analysis, the authors rely on pre-existing curated collections of economic documents, instead of trying to identify latent statistical features such as topics that feature economic terms, or documents that contain words with similar semantics to specific keywords. We believe that their analysis would be improved by augmenting their information retrieval capacities, as well as view the change of specific words over time.

4 Methods

4.1 Corpus

The Early English Books Online (EEBO) collection contains around 125,000 digital facsimiles of virtually all printed works in English-speaking countries from 1473-1700. The EEBO Text Creation Partnership (EEBO-TCP) is a project to manually turn the digital facsimiles into machine-readable text. There are over 40,000 texts publicly available for download, and are available alongside basic summary statistic tools.

4.1.1 Challenges

Most text processing packages require text to be encoded via Unicode. However, in order to represent the original documents with the highest accuracy, the EEBO-TCP XML corpus uses non-Unicode where they are found in the original documents. In order to circumvent this issue, we used the 'unidecode' package, which replaces characters with their closest resembling ASCII equivalent.

Spelling consistency is critical for probabilistic language models to work correctly. Unfortunately, early English orthographies at the advent of the printing press was almost entirely up to the discretion of the printer. Consider the philosophy of the first printer of English, William Caxton:

> While his English was clearly based on the emerging standard language of London, Caxton's approach to spelling does not constitute a concerted attempt to create a standard. His spelling varies widely within each book, and even more from book to book. [13]

It took hundreds of years for English spelling to become more regular across print media, with many well-funded and authoritative efforts to introduce and enforce spelling norms falling by the wayside. Only until the late 19th century were some of Noah Webster's spelling suggestions were integrated, and only in American English, leading to more divides, now between American and British spelling.

This presents serious problems for anyone trying to work on English before the 19th century with computational methods. During training, a model will tokenize words based on their spelling, so 'goodnesse' and 'goodness', both referring to the same entity, will register as different tokens. [2] These problems confound the ability of a model to accurately learn the semantics of either token.

While spelling error detection and correction is common today for current English, these methods cannot immediately be applied to early English, as the kinds of errors that these models are designed to correct (various typos for a single word) aren't the same as multiple correctly spelled words. The only major tool for dealing with early English spelling variation is Variant Detector (VARD), which is not open-source, and not publicly available without the permission of the authors. [3] As a result, we resorted to using a collection of dictionary-based replacement methods, where words with potential variations were located in a dictionary, and replaced with their entry.

4.1.2 Implementation

We used a combination of pipelines in order to process the EEBO-TCP TEI encoded documents into cleaned plain-text documents and lists of tokens, also known as Bag of Words (BoW). First, we used the University of Wisconsin-Madison's Visualizing English Print (VEP) pipeline to preprocess the TEI XML files from EEBO into plaintext, replaced special EEBO characters with unicode substitutes, and finally uses a spelling correction dictionary to normalize word spellings. Then, we used additional spelling correction dictionaries compiled by Texas AM's Early Modern OCR Project to replace spelling variations, OCR errors, and syncopes in the VEP-processed corpus. Finally, for Bag of Words representations of the corpus, we used the NLTK WordNet Lemmatizer to lemmatize words, for example replacing 'feet' with 'foot', which improves the performance of topic models.

4.2 Dynamic Word Embeddings

4.2.1 Word Embeddings

In text analysis, being able to retrieve information from documents or collections of documents is critical. However, simple keyword searches do not represent the richness and relations between vocabulary words in a language. Consider, for example, that while "dog" and "cat" are both pets, a simple keyword search for "pets" in a document will not return them as results. Here, the relationship between these words isn't adequately represented; there needs to be a better way to represent words and vocabularies.

This inability to represent word relationships is because keyword searches represent each word in a vocabulary as a *local* representation. In a local representation, each word in the vocabulary is given an index. Thus, every word can be considered a boolean vector with dimension the size of the vocabulary, which is true in the index assigned to it and false everywhere else. For example, if we have a set of cars which include {"large white Toyota", "small black Ford", "large black Toyota", "small white Ford"}, each car would would get its own index, despite clearly sharing comparative features, and so the set of vectors would be $\{[1000], [0100], [0010], [0001]\}$, which have dimension $D = 4$. Consider that the dot product between any two words, the cosine similarity between two vectors, would be 0. If we consider the matrix of locally represented vectors, we can see that it is extremely sparse, and thus encodes little information about the words it aims to represent. Ideally, we want to reduce the dimensionality of this matrix from a $|V| \times |V|$ to a $|V| \times D$, where D is significantly less than the vocabulary size. This forces the vectors to be less sparse, which in turn encodes more information in the vectors.

The alternative to a local representation of words is a *distributed* representation, which encode relationships between elements of the sets. If we use the list of cars again as an example, we can use "size", "color" and "brand" as the categories which the elements are distributed across, which would lead to encodings {[1 1 0], [0 0 1], [1 0 0], [0 1 1]}. Note that these representations are denser than the local representations, since they have dimension $D = 3$. Furthermore, we can extract latent features within the vocabulary by performing operations on these vectors. For example, the degree of commonality between a "large white Toyota" [110] and a "small white Ford" [011] is the dot product of their distributed representations (1, corresponding to them both being white). Here, car information is preserved in the representations of elements as vectors in low-dimensional vector spaces, a "car space".

The principle of distributed representations of cars extends easily to words, where instead of car information, semantic meaning is preserved in the vector representations, which we refer to as word embeddings in a "word vector space". Embeddings rely on the Distributional Hypothesis, which essentially states that the meaning of words can be derived from the contexts that they appear in. Thus, by assembling large samples of words and the contexts that they appear in, we should be able to extract an embedding for every word in the vocabulary.

There are a variety of methods to assemble word embeddings for a vocabulary given set of documents. One set of methods involves creating a co-occurrence matrix of words, where each row and column is a word and each index is the number of times those words appear together in context, and to then factorize this matrix into lower dimensions. [8] More modern approaches are probabilistic, which rely on neural networks to model the relationship between word-context pairs. [15] For example, given a sentence like "the quick brown fox jumps over the fence", the set of word-context pairs would include {(["the", "quick", "fox", "jumps"], "brown"), (["quick", "brown", "jumps", "over"], "fox"), (["brown", "fox", "over" "the"], "jumps"), ...}. The neural network then tries to maximize the likelihood of outputting the context words given a target word, or vice versa,

depending on the model. Generally, probabilistic word embeddings continue to set and achieve benchmark performances on a number of natural language tasks, including translation, analogy puzzles, and question-answering. [4]

Surprisingly, word embeddings preserve semantic structure in arithmetic operations. For example, `vec`("king") - `vec`("man") + `vec`("woman") is most close to the vector for "queen". This suggests that word embeddings preserve similarity, and encode it geometrically, which we can then leverage visually to create plots of words and their closest neighbors, for example.

4.2.2 Dynamic Word Embeddings

Dynamic word embeddings aim to incorporate the semantic drift of words by dividing a corpus into a set of time slices and forming the word embeddings for each time period. By creating dynamic word embeddings, we can quantify linguistic features and how they change over time.

They base their method on the work by Levy and Goldberg [12] that skip-gram negative sampling (SGNS) word embeddings, one of the most popular methods for creating word embeddings, are equivalent to the implicit weighted factorization of the pointwise mutual information matrix, a common metric in information theory for measuring the association between a pair of discrete outcomes, into a set of word embeddings and context embeddings. This in itself is a beautiful result, that neural probabilistic models for word embeddings are delicately tied to the original matrix factorization methods that were first proposed. Levy and Goldberg further show that by using the positive pointwise mutual information (PPMI) matrix for factorization, they can improve performance on certain common language tasks.

Yao et al [25] approach the problem of semantic drift by dividing a corpus into time periods and calculating the PPMI over all the documents in each time period. They then create the word and context embeddings from the PPMIs by minimizing a cost function with three terms: a reconstruction loss, which implicitly factorizes the matrix; an L_2 norm which regularizes the embeddings; and an alignment cost that forces word vectors from neighboring times to be similar to each other.

4.2.3 Implementation

For our initial static embeddings, we chose to learn FastText embeddings, which use n-grams to incorporate sub-word information into the embeddings, thus more effectively dealing with spelling variation. [7] We then rewrote the original authors' implementation of the Dynamic Word Embeddings algorithm to use PyTorch, a machine learning library that optimizes the matrix operations during training. Due to time-complexity issues, we approximate the word-cooccurence matrix rather than calculate it directly. We train the DWE model using the default parameters.

4.3 Dynamic Topic Models

4.3.1 Topic Modeling

The problem of being able to organize a collection of documents is fundamental to any library of text. In the past, journals and newspapers like Science have manually tagged each document with the set of topics that are present. However, doing this by hand is impossible for existing, untagged volumes. Thus, automatically identifying sets of words which correspond to topics, and tagging sets of documents according which topics they represent is a field of information retrieval with a long history.

The first popular topic model was the tf-idf scheme [20], which assembles an DxV matrix, where D was the number of documents in the collection, V was the number of words in the vocabulary, and each element in the matrix corresponds to the number of times the vth word occurs in the dth document. Each column in the matrix is then normalized by the number of appearances of the vth word in the entire corpus; thus, documents with uncommon words have high values in their corresponding column. By representing each document as a V-length vector, we can group similar vectors to identify topics.

While tf-idf schemes worked at identifying groups of documents, they don't compress or summarize the contents well, as each document is still V-dimensional. The natural extension of the matrix approach to tf-idf is to project it down to lower dimensions via singular value decomposition; this method is referred to as latent semantic indexing (LSI). [8]

Both tf-idf and LSI represent document and topic structures well, but they lack a probabilistic motivation; that is, they don't provide motivation as to why they should work or how documents are constructed. Probabilistic Latent Semantic Indexing (pLSI) aims to address this by treating each document as a mixture model of topics, where each word comes from a single topic. [11] Each document is a list of topics and their corresponding mixing proportions, and each topic is a multinomial distribution over words. However, this only incorporates a probabilistic construction of topics, not documents.

In working towards a probabilistic model of both words and documents, we assume exchangeability: the order of the words doesn't matter (also known as a Bag of Words assumption), and the order of documents doesn't matter. By assuming exchangeability, we can greatly simplify the calculation of the probability function that measures the likelihood of a document given a set of learned parameters, which we seek to maximize. Greatly simplifying the calculation of learned parameters allows us to construct an entirely probabilistic description of documents and topics computationally tractable, and perform inference over the parameters.

One algorithm which exploits exchangeability is Latent Dirichlet Allocation (LDA) .[6] LDA assumes that each document is constructed via the following procedure:

1. Draw $N \sim \text{Poisson}(\xi)$, the number of words in the document.

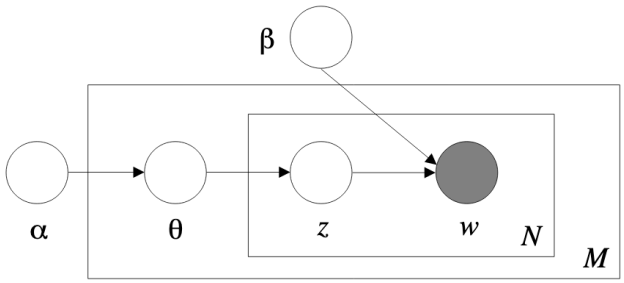

Figure 1: Plate diagram for LDA.

2. Draw $\theta \sim \text{Dirichlet}(\alpha)$, the multinomial parameter dictating the mixture of topics, or the document-topic distribution, where α is a hyperparameter used to represent the sparsity of document-topic distributions.
3. For each word n in N:
   (a) Draw $z_n \sim \text{Multinomial}(\theta)$, a topic assignment for the nth word
   (b) From the β_z topic, draw $w_n \sim \text{Multinomial}(\beta_z)$, the word-topic distribution.

Each β_z is drawn from a Dirichlet distribution η, which represents the sparsity of the word-topic distribution. Performing this procedure over M documents is how we construct our corpus.

4.3.2 Dynamic Topic Modeling

Unlike in LDA, in order to capture the change in topics over time, we cannot make the same exchangeability of documents assumption, since the documents appear in chronological order. In Dynamic Topic Modeling (DTM) [5], we have a given set of time slices, and each time slice should have its own set of word-topic distributions that words are drawn from. Therefore, instead of using Dirichlet distributions to draw word-topic distributions from, we instead model the word-topic distribution parameter as a V-dimensional vector which evolves due to Gaussian noise, and redraw the vector at each time slice.

The new procedure for generating documents is as follows:

1. Draw topic $\beta_{t,i}$ for topic i at time t $\beta_{t,i}|\beta_{t-1,i} \sim N(\beta_{t-1,i}, \sigma^2 I)$.
2. Draw the sparsity α_t at time t $\alpha_t|\alpha_{t-1} \sim N(\alpha_{t-1}, \delta^2 I)$.
3. For each document:
   (a) Draw $\eta \sim N(\alpha_t, a^2 I)$, the unnormalized document-topic distribution.
   (b) For each word:
      i. Draw $z \sim \text{Mult}(\pi(\eta))$, where $\pi(\eta)$ is a normalizing function.

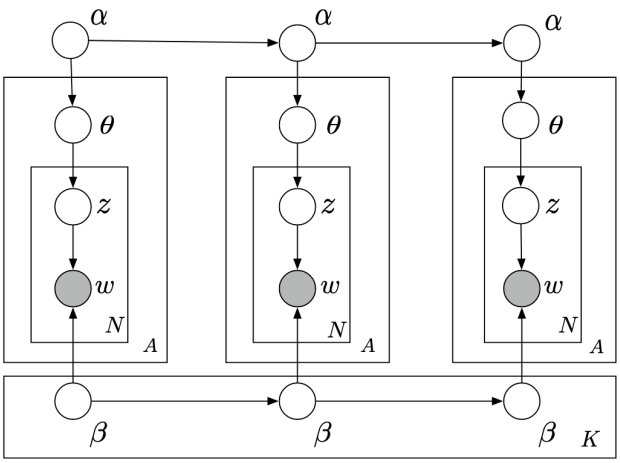

Figure 2: Plate diagram for DTM.

ii. Draw $w \sim \text{Mult}(\pi(\beta_z))$, where $\pi(\beta)$ is a normalizing function.

We can then see the evolution of topics as a change in the multinomial distribution over the vocabulary.

4.3.3 Implementation

We rely on the Gensim wrapper for the original Dynamic Topic Model implementation by the authors. We train using the default parameters.

5 Experiments

5.1 Dynamic Word Embeddings

5.1.1 Proximity

By computing the dot product between a given word embedding and all of the embeddings in the vocabulary, we can compute the words with the closest meaning over each time slice. [21]

5.1.2 Projection

Given a two words, we are able to project one vector onto another, which is a measure of their semantic correlation or overlap. In a simple projection, we compare how well a given word and a set of words correlate over time. [21]

In a complex projection, we project a set of vectors onto the difference between two vectors, thus mapping words onto a semantic spectrum. [21]

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
english	english	exercise	trade	trade	trade	trade	trade	trade	trade	trade	trade	trade	trade	trade	trade	trade
slept	slept	english	exercise	gain	merchants	merchants	merchants	commodities	merchants	commodities	money	goods	goods	colonies	commerce	commerce
awoke	saxons	merchants	christian	merchandise	merchandise	commodities	commodities	merchants	commodities	merchants	foreign	money	province	commerce	goods	states
dragon	waking	looking	religion	merchants	merchant	merchandise	traffic	traffic	money	money	commodities	france	foreign	goods	foreign	articles
waking	exercise	forward	reading	exercise	commodities	traffic	merchandise	merchandise	foreign	foreign	goods	manufactures	manufactures	manufactures	manufactures	goods
window	modo	trade	study	wares	gain	merchant	merchant	money	merchandise	goods	merchants	commerce	money	america	colonies	united
latin	normans	merchandise	diligent	merchant	sell	sell	money	countries	goods	manufactures	manufactures	england	colonies	british	states	manufactures
drowned	danes	wares	diligently	honest	wares	buy	sell	merchant	traffic	ships	england	country	england	commodities	duties	article
leapt	changed	reading	forward	travail	buy	money	sold	foreign	merchant	countries	nation	french	commerce	majesty	america	foreign
sleep	hostess	merchant	teaching	study	countries	sold	buy	towns	countries	merchant	english	merchants	commodities	foreign	articles	british

Figure 3: Word embedding for 'trade' and ten nearest neighbors. 'Trade' is not at the top of all of the columns because at earlier times, the word was not present in the corpus.

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
body	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies	bodies
bodies	body	body	body	body	body	body	body	body	air	spirits	motion	body	animal	motion	fluid	animal
buried	buried	buried	flesh	flesh	flesh	soul	spirits	spirits	body	air	body	animal	body	animal	animal	air
soul	fire	soul	soul	soul	soul	flesh	flesh	natural	spirits	motion	spirits	blood	air	body	heat	heat
fire	hell	flesh	natural	dead	spiritual	spirits	natural	flesh	motion	body	particles	earth	motion	heat	motion	animals
earth	earth	fire	fire	earth	earth	natural	spiritual	motion	heat	particles	earth	spirits	heat	air	particles	atmosphere
rome	divers	divers	spiritual	natural	dead	souls	soul	air	substance	substance	air	motion	particles	particles	air	body
hell	soul	deed	earth	spiritual	natural	spiritual	nature	heat	motions	natural	parts	flesh	blood	blood	animals	motion
places	diverse	hell	dead	nature	substance	nature	creatures	cold	natural	animal	substance	air	earth	earth	surface	substances
four	deed	places	nature	souls	fire	earth	souls	spiritual	cold	heat	animal	natural	qualities	fluid	body	surface

Figure 4: Word embedding for 'bodies' and ten nearest neighbors.

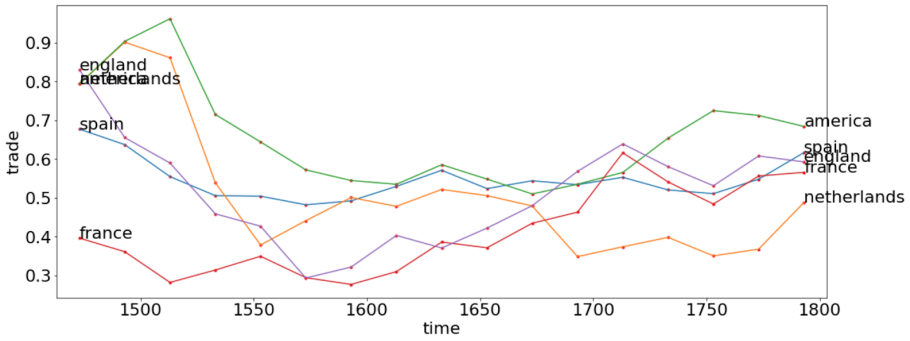

Figure 5: Simple projection of various words onto 'trade'.

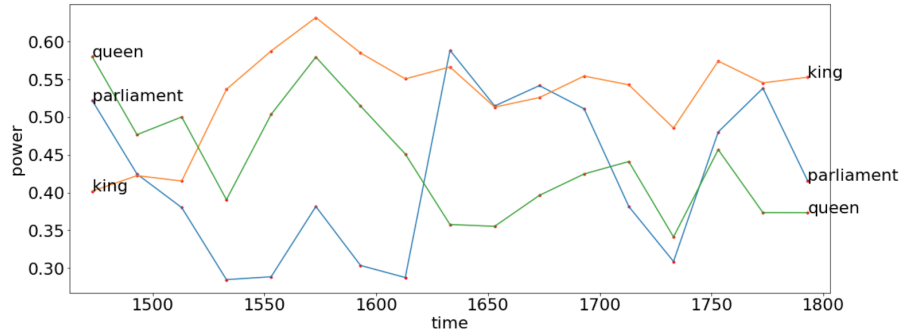

Figure 6: Simple projection of various words onto 'power'.

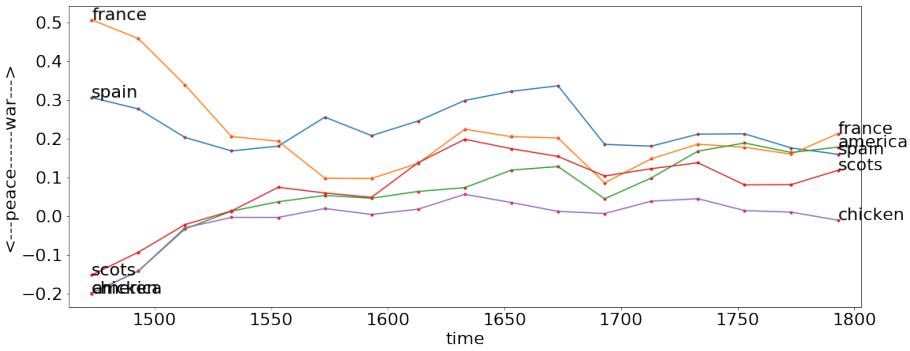

Figure 7: Complex projection of various country names onto the spectrum of 'war' and 'peace', where 'chicken' is a control.

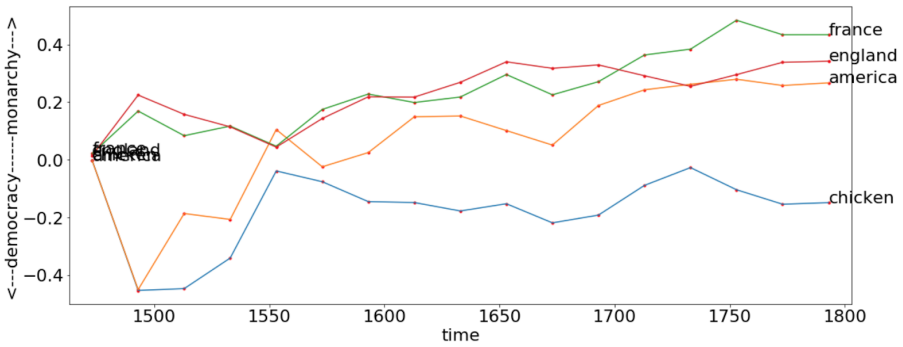

Figure 8: Complex projection of various country names onto the spectrum of 'monarchy' and 'democracy', where 'chicken' is a control.

5.1.3 Rejection

We are able to disambiguate polysemic embedding features by using vector rejections to remove unwanted additional meanings from words. [21]

5.1.4 Norms

The embedding norm correlates with the frequency of usage. [25]

5.2 Dynamic Topic Modeling

6 Discussion

We seek to evaluate how well dynamic word embeddings and dynamic topic models reflect known historical trends.

In Figure 4, we can see that the nearest neighbors 'Bodies' changes from religious to scientific. This corresponds neatly to the historical record: the European Enlightenment begins around 1637, and Newton's Principia Mathematica is published in 1687. Until 1633, the nearest neighbors are mainly religious in nature. Then, from 1633 to 1673, they become pseudoscientific. Finally, from 1673 onwards, they take on the connotations of 'heavenly bodies', and seem to be used to describe physical observations.

In Figure 7, we plot vectors along the 'war-peace' spectrum. In the historical record, England went to war with Spain in 1568 again in 1655, which is where the blue line peaks. 'america' steadily rises in hostility. 'chicken' is used as a control group, and stays below all of the actual countries England was at war with.

In Figure 6 Cosine similarity between 'power' and 'king', 'queen' and 'parliament'. From 1553 to 1603, England was ruled by Queen Mary and Queen Elizabeth. In 1604, Britain was unified after the union of the English and Scot-

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
river	le	river	river	river	river	river	river	river	river	river	river	corn	water	water	water	river
mile	en	cest	mile	water	water	mile	mile	mile	mile	mile	corn	ground	river	river	river	water
hylle	ou	les	west	mile	mile	water	west	west	west	soit	mile	sleep	corn	mile	side	stone
water	les	sil	water	west	hills	west	hills	side	side	luy	ground	sell	stone	stone	stone	mile
runs	ill	roy	north	north	side	side	water	north	south	lez	sell	money	ground	foot	fro	side
hills	luy	vne	south	hills	west	north	ryuers	hills	ryuers	mez	half	river	foot	ground	thou	sleep
drowned	lez	hylle	hills	side	ryuers	hills	north	ryuers	north	ryuers	sleep	weight	weight	tree	sleep	fro
mount	fait	jour	ryuers	hylle	north	south	south	south	hills	tenant	west	sold	half	slept	hath	ground
deep	pur	ryuers	stretches	south	south	ryuers	side	stretches	corn	les	foot	silver	sell	side	thy	foot
ryuers	soit	estre	hylle	ryuers	hylle	hylle	stretches	water	stretches	cest	weight	paye	mile	drowned	ground	therthe

Figure 9: Word embedding for 'bank' and nearest neighbors.

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
vne	le	sil	vne	vne	auxi	bridge	bridge	yearly	pitem	soit	corn	money	pens	pens	money	paye
jour	en	deuant	deuant	nest	coo	trees	early	payment	corn	mez	pens	paye	silver	silver	paye	money
deuant	ill	vne	cel	sil	case	ryuers	ryuers	grass	yearly	tenant	paye	pens	corn	stone	silver	silver
ie	ou	jour	sil	auxi	nest	bows	tide	woods	pay	luy	paid	silver	money	penny	paid	penny
vous	les	bon	celuy	mort	celuy	waters	feldes	groude	paye	lez	money	sold	sold	slept	pens	pens
bien	de	cest	cell	deux	auera	early	slept	watch	pens	aver	sell	paid	weight	weight	penny	paid
roy	luy	cel	auxi	celuy	sot	hylle	grass	bridge	payment	ascun	value	weight	penny	whet	weight	stone
sil	lez	estre	nest	cell	cell	lodged	woods	slept	rent	issint	sold	worth	sell	half	sold	sold
dieu	tiel	tout	deux	deuant	mort	tide	flowers	pens	paid	qe	weight	corn	whet	pound	sleep	weight
justices	fait	auxi	mort	sot	aver	pond	waters	chapel	term	fee	penny	penny	worth	money	stone	sleep

Figure 10: Word embedding for 'bank' rejected from 'river' and ten nearest neighbors.

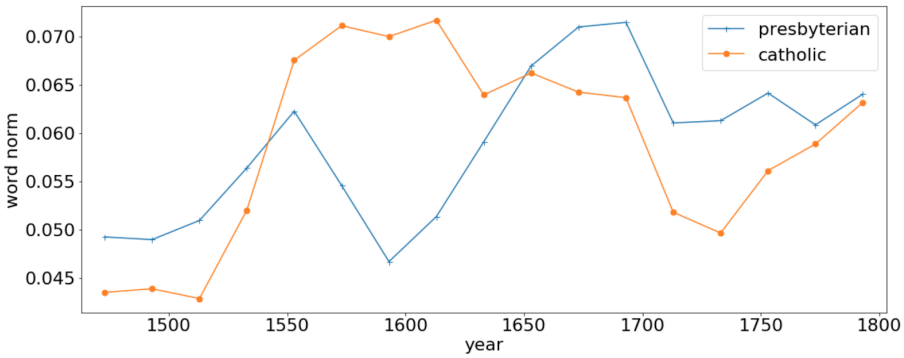

Figure 11: Word norms of 'presbyterian' and 'catholic'.

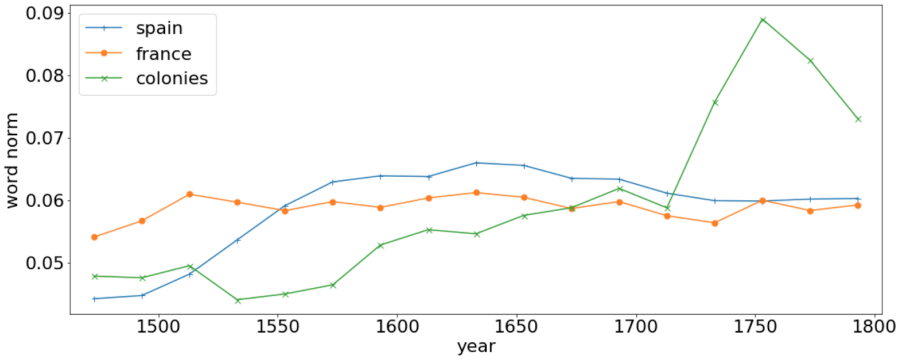

Figure 12: Word norms of 'spain', 'france', and 'colonies'.

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
ill	ill	realm	realm	person	law	king	king	king	king	king	king	law	law	law	state	state
person	person	person	person	realm	king	law	law	law	parliament	law	law	king	act	colony	government	government
realm	realm	law	law	law	majesty	majesty	majesty	parliament	law	parliament	parliament	person	person	government	law	law
heir	heir	king	king	king	person	person	parliament	majesty	majesty	majesty	majesty	act	government	act	act	united
soit	law	majesty	majesty	majesty	realm	common	person	common	person	person	act	parliament	court	person	great	country
manner	manner	manner	manner	cause	common	court	common	person	court	act	person	majesty	province	court	right	great
law	king	ill	say	common	cause	subject	court	court	common	court	court	court	colony	state	people	people
tenant	majesty	heir	cause	subject	subject	parliament	subject	kingdom	act	common	time	time	time	great	country	right
home	soit	say	subject	say	court	cause	kingdom	house	house	time	england	government	parliament	time	power	power
king	say	cause	common	manner	time	realm	time	time	kingdom	house	common	england	king	right	new	general

Figure 13: An example topic from the DTM results. Here, the topic is the evolution of law.

1473	1493	1513	1533	1553	1573	1593	1613	1633	1653	1673	1693	1713	1733	1753	1773	1793
word	word	word	word	word	word	latin	latin	latin	latin	latin	latin	epistle	epistle	epistle	epistle	epistle
english	english	english	translate	translation	latin	word	alphabet	alphabet	alphabet	alphabet	alphabet	latin	word	paul	paul	paul
translate	translate	translate	english	translate	translation	alphabet	word	word	word	word	word	word	paul	word	word	word
verb	verb	translation	translation	english	translate	translation	chrysostome	jewes	render	render	epistle	alphabet	latin	latin	verb	sentence
translation	translation	verb	latin	latin	english	english	english	verb	jewes	epistle	render	paul	alphabet	verb	sentence	verb
latin	latin	latin	verb	verb	alphabet	translate	translation	signify	epistle	verb	verb	render	verb	sound	latin	sound
greek	greek	greek	greek	greek	verb	chrysostome	verb	english	paraphrase	signify	viz	verb	sound	sentence	sound	latin
text	text	text	text	text	text	verb	signify	epistle	signify	viz	paul	viz	render	alphabet	express	noun
signification	signification	bible	bible	alphabet	greek	text	phrase	chrysostome	verb	style	signify	eusebius	eusebius	eusebius	noun	syllable
tongue	bible	signification	signification	bible	bible	descend	text	phrase	style	notion	grotius	grotius	viz	express	syllable	mac

Figure 14: An example topic from the DTM results.

1533	1553	1573	1593	1613
steorte	steorte	steorte	steorte	steorte
warnborne	warnborne	warnborne	warnborne	warnborne
uuinsla	uuinsla	uuinsla	uuinsla	uuinsla
uessci	uessci	uessci	uessci	uessci
hannyngfeld	hannyngfeld	hannyngfeld	hannyngfeld	hannyngfeld
bonewood	bonewood	bonewood	bonewood	bonewood
dfrois	dfrois	dfrois	dfrois	dfrois
colyngborne	colyngborne	colyngborne	colyngborne	colyngborne
monsichet	monsichet	monsichet	monsichet	monsichet
bedwynd	bedwynd	bedwynd	bedwynd	bedwynd

Figure 15: Subset of topic 1 from a DTM with 150 learned topics.

tish crowns, which "increased parliamentary authority at the expense of royal authority." [23]

In Figures 9 and 10, we not only see that the rejection of 'river' from 'bank' yields more economical terms than 'bank' alone, but we also see a close congruency with the historical record: the pecuniary terms start in 1693, and the Bank of England was founded in 1694.

In Figure 12, while 'france' and 'spain' stay relatively placid, the norm for 'colonies' increases rapidly during the period of British colonial unrest and subsequent revolution.

In Figure 11, the rise of the norm of the term 'presbyterian' corresponds to the date of the founding of the Presbyterian church in 1646. Similarly, the Church of England is established in 1534, when Henry VIII split from the Catholic Church to annul his marriage.

In Figures 13 and 14, the results of two coherent topics from a dynamic topic model trained on 150 topics, we see that they seem to reflect a semantic shift from start to end. While the evolution in 13 can be accounted for by the transition from divine right to parliamentary representation, the evolution in 14 is harder to explain. Did the topic of English translation really evolve into Christian scholarship? It is harder to qualitatively evaluate the association between dynamic topics and the historical record, since the analysis is a question of how the literature changed, which is almost tautological given the question.

Finally, we believe that dynamic topic models are too sensitive to hapax legomena (words used rarely or once in the whole corpus). For example, consider Figure 15, which presents a fragment of a topic in a learned dynamic topic model with 150 topics. Of the 150 learned, 97 of them contained hapax legomena such as 'steorte' or 'hannyngfeld'. While the remaining 53 topics gave relatively compelling results, the model as a whole seems unable to incorporate word frequency as a metric when learning the word-topic distributions.

7 Future Work

7.1 Dynamic Topic Models

We hypothesize that while they are good at maintaining minimal between-topic overlap, thereby creating distinct topics, the word-topic distribution vectors move around in the latent space to locations, which are too distant from their starting positions. This causes the each topic to make a semantic jump over the course of its evolution, from one set of words to another, unrelated set. We thus suggest imposing a per-topic semantic drift term in the objective function, which would penalize derived topics from linking two separate human topics.

In addition, we also suggest adapting unbounded topic modeling via neural variational inference while jointly learning word embeddings, to analyze the emergence and disappearance of topics over time, which we outline in the appendix.

Finally, given the model's proclivity to fill topics with hapax legomena, we see a need to make the model more robust to poorly cleaned corpuses. While more exhaustive corpus cleaning may be an effective solution, LDA is supposed to be agnostic to hapax legomena, thus a dynamic topic model should also preserve this insensitivity. [18]

7.2 Dynamic Word Embeddings

We stress the difficulty of working on corpuses with high orthographic variation. While cleaning the corpus, we used a hand-labeled correction dictionary of 100,000 rules to replace tokens in documents. For our initial embeddings, we used FastText embeddings, which incorporate sub-word information when constructing embeddings. However, a better approach might be to algorithmically identify and correct spelling errors in the texts, perhaps via recurrent neural networks. While VARD 2 exhibits strong precision and recall, is based on historical spelling trends, and was developed by relevant experts in the field, it is not publicly available. We thus suggest a more naive model of adapting the dictionary correction rules as training data for a language model capable of identifying and correcting spelling variations.

In addition, based on [1], word similarity measures between embeddings, the foundation of our suite of inquiries, are not stable. This may result in different orderings of the nearest word neighbors, and in general will result in variability across all of the results we presented here. To obtain reliable results, similarity measures should be bootstrapped over multiple individual embeddings of the same corpus.

7.3 Digital Humanities

More generally, we believe that the narrative of English semantic drift can be more fruitfully studied by extending the range of the corpus from beyond 1800 to the current day. Given the exponential increase of the corpus size in time, we

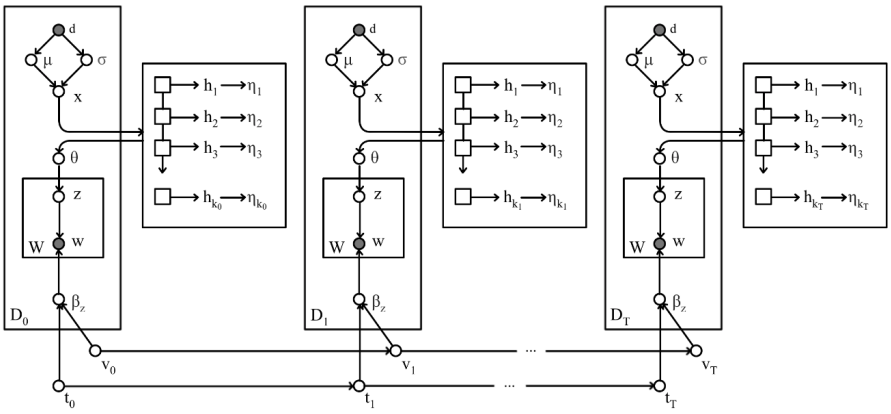

Figure 16: Plate diagram for UDNTM.

suspect that these models and subsequent methods of inquiry may be sensitive to unequal distributions of documents in each time slice.

8 Conclusion

Overall, we believe both Dynamic Topic Models and Dynamic Word Embeddings show promise for researchers interested in exploring early modern English texts. Dynamic Word Embeddings appear to successfully encode the cultural contexts of words and their changes over time. Dynamic Topic Models are sensitive to corpus quality, especially with respect to spelling variations and hyperparameter tuning.

A Unbounded Dynamic Neural Topic Models

Here, we present a brief outline of an unbounded dynamic neural topic model, which jointly learns topics and embeddings, as well as dynamically learning the number of topics present at each time slice, based on [14].

First, we divide the corpus into a set of time slices, $D_1, .., D_T$, and initialize the number of topics as k_0. For each document in the time slice, $d \in \mathrm{R}^{|V|}$ is a Bag of Words (BoW) representation of the document. $\mu, \log \sigma \in \mathrm{R}^H$ are the variational parameters of the latent document space, and are the output of a multi-layer perceptron MLP(d). The latent document vector is drawn from the latent document space, $x \sim N(\mu, \sigma^2 I) \in \mathrm{R}^H$.

We then pass x to a recurrent neural network in order to determine the topic distributions present, via the recurrent stick-breaking construction. Given k as the number of topics, $h_k = \mathrm{RNN}(h_{k-1}) \in \mathrm{R}^H$, the output of the kth state, and $\eta_k = \mathrm{sigmoid}(h_{k-1}^t x)$. $\eta \in \mathrm{R}^k$ is then passed to a stick-breaking function f_{SB},

which maps η to $f_{SB}(\eta) = \theta \in \mathrm{R}^k$ on the $k-1$-dimensional simplex. θ is the multinomial parameter for the document-topic distribution.

$t_i \in \mathrm{R}^{k \times H}$ is the latent topic matrix at the ith time slice, where the jth row is the latent topic vector for the jth topic. $v_i \in \mathrm{R}^{|V| \times H}$ is the word embedding matrix at the ith time slice, where the jth row is the embedding for the jth word in the vocabulary. $\beta_i = t_i v_i^T \in \mathrm{R}^{k \times |V|}$ is the word-topic matrix at time slice i, and $\mathrm{softmax}(\beta_{i,j})$ is the multinomial parameter for the jth word-topic distribution at time i.

Given θ, the parameter of the document-topic multinomial, we draw $z_w \sim \mathrm{Mult}(\theta)$, and subsequently $w \sim \mathrm{Mult}(\beta_z)$ for every word $w \in D$.

We then calculate the variational lower bound for the document with k topics and N words:

$$\mathcal{L}_d^k = \sum_{n=1}^{N} \log p(w_n | \beta^k, \theta^k) - D_{KL}(q(x|d)||p(x)),$$

where $D_K L$ is the KL Divergence between the posterior and the prior.

In order to dynamically determine the number of topics in each time slice, we calculate the likelihood increase brought by adding the additional ith topic across all the documents in D_i:

$$\mathcal{I} = \frac{\sum_{d \in D_i} \mathcal{L}_d^k - \mathcal{L}_d^{k-1}}{\sum_{d \in D_i} \mathcal{L}_d^k}.$$

If $\mathcal{I}$ is greater than an acceptance hyperparameter γ, then we increase the active number of topics. Upon determining the updated number of topics, we chain the latent word and topic matrices to the next time slice via the same Gaussian noise process in [5].

Inference over the generative parameters Θ, including t, v, and the RNN, as well as for the variational parameters Φ, including $\mu(d)$ and $\sigma(d)$, are jointly updated via stochastic gradient backpropogation on the variational lower bound.

References

[1] Maria Antoniak and David Mimno. "Evaluating the stability of embedding-based word similarities". In: *Transactions of the Association of Computational Linguistics* 6 (2018), pp. 107–119.

[2] Alistair Baron. "Dealing with spelling variation in Early Modern English texts". PhD thesis. Lancaster University, 2011.

[3] Alistair Baron and Paul Rayson. "VARD2: A tool for dealing with spelling variation in historical corpora". In: *Postgraduate conference in corpus linguistics*. 2008.

[4] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vol. 1. 2014, pp. 238–247.

[5] David M Blei and John D Lafferty. "Dynamic topic models". In: *Proceedings of the 23rd international conference on Machine learning*. ACM. 2006, pp. 113–120.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[7] Piotr Bojanowski et al. "Enriching word vectors with subword information". In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146.

[8] Scott Deerwester et al. "Indexing by latent semantic analysis". In: *Journal of the American society for information science* 41.6 (1990), pp. 391–407.

[9] A S G Edwards. *William Caxton and the introduction of printing to England*. 2018. URL: `https://www.bl.uk/medieval-literature/articles/william-caxton-and-the-introduction-of-printing-to-england`.

[10] Emily Erikson and Mark Hamilton. "Companies and the Rise of Economic Thought: The Institutional Foundations of Early Economics in England, 1550–1720". In: *American Journal of Sociology* 124.1 (2018), pp. 111–149.

[11] Thomas Hofmann. "Probabilistic latent semantic analysis". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. 1999, pp. 289–296.

[12] Omer Levy and Yoav Goldberg. "Neural word embedding as implicit matrix factorization". In: *Advances in neural information processing systems*. 2014, pp. 2177–2185.

[13] The British Library. *Caxton's Chaucer*. URL: `https://www.bl.uk/treasures/caxton/prtcaxenglish.html`.

[14] Yishu Miao, Edward Grefenstette, and Phil Blunsom. "Discovering discrete latent topics with neural variational inference". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 2410–2419.

[15] Tomas Mikolov et al. "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*. 2013, pp. 3111–3119.

[16] Matthew Milner, Stephen Wittek, and Stefan Sinclair. "Introducing DREaM (Distant Reading Early Modernity)". In: *DHQ* 11.4 (2017).

[17] David Mitch. "Education and skill of the British labour force". In: *The Cambridge economic history of modern Britain* 1 (2004), pp. 1700–1860.

[18] Radim Rehurek. *Is it a must to remove tokens$_o$ncewhenbuildingaLDAmodel*?. 2013. URL: https://groups.google.com/forum/?hl=hy#!topic/gensim/cjgJottKGSI.

[19] Max Roser. "Books". In: *Our World in Data* (2019). https://ourworldindata.org/books.

[20] Gerard Salton and Christopher Buckley. "Term-weighting approaches in automatic text retrieval". In: *Information processing & management* 24.5 (1988), pp. 513–523.

[21] Ben Schmidt. *Word Embeddings for the digital humanities*. 2015. URL: http://bookworm.benschmidt.org/posts/2015-10-25-Word-Embeddings.html.

[22] *Text Creation Partnership*. URL: https://www.textcreationpartnership.org/.

[23] Charles Upchurch. *A Timeline of Modern English History*. URL: http://myweb.fsu.edu/cupchurch/Resources/Timeline_ModBrit.html.

[24] Melvin Wevers, Tom Kenter, and Pim Huijnen. "Concepts through time: tracing concepts in Dutch Newspaper Discourse (1890–1990) using word embeddings". In: *Digital Humanities* 2015 (2015), p. 1.

[25] Zijun Yao et al. "Dynamic word embeddings for evolving semantic discovery". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM. 2018, pp. 673–681.